


Upgrade Notes 2.0

When upgrading Embedded Engineer from 1.2 to 2.0, please check your model as there have been incompatible changes between these versions.

 You can install Embedded Engineer 1.2 and 2.0 side by side. It's recommended to do so in the transition period, making sure your model plays nice with the new version of Embedded Engineer.

Signals Enumeration

In Embedded Engineer 1.2, the [signals enumeration](#) had to be generated explicitly, and wasn't available in the model. Code generation parsed the whole package for which the enumeration was generated, and added each signal to the enumeration, effectively doing a code generation step on its own.

With Embedded Engineer 2.0, the signals enumeration is now part of the model. This has many benefits, as you can now add custom entries to it, change the order of enumeration entries, set values for the enumeration entries and so on.

When you upgrade your model from 1.2 to 2.0, add an enumeration [as described in this help](#), and at least generate the signals once.

No automatic semicolons

Embedded Engineer 1.2 automatically added semicolons at the end of effects and actions. With Embedded Engineer 2.0, this behavior was changed to *not* add such semicolons. Unfortunately the old behavior clashed with the new [Code Synchronization](#) feature, so it had to be dropped.

New code formatting

Previously, an open source tool named [Artistic Style](#) has been used to format the generated source code. While it was possible to exchange this tool with another one, Embedded Engineer now requires advanced features from the code formatting tool, which Artistic Style unfortunately doesn't provide. Faced with the difficult decision of either advancing Artistic Style to the point where we can use it in Embedded Engineer, or to use another tool, we decided on the latter. We found an able successor in the tool suite [ClangFormat](#) which covers most of the formatting functionality we need, and plays nice with custom formatting steps required for certain coding style guidelines.

Because of the new code formatting tool, code will be formatted differently as before.

Return parameter of activities

In Embedded Engineer 1.2, the return parameter of activities had to be marked as *Out* parameter. In the new version, it has to be [marked as Return parameter](#), making the intent clear.

Activity parameters of type *InOut*

Activity parameters with *InOut* direction are now generated with pointer semantics. This may break functionality with (albeit previously invalid) models.